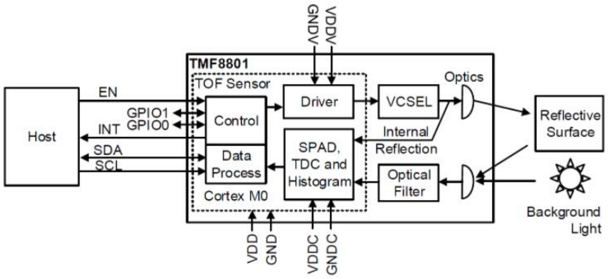
SENSORS CONVERGE – THE AMS OSRAM TMF 8801 TIME-OF-FLIGHT SENSOR

Introduction

The TMF8801 Time-of-flight (ToF) sensor from ams enables highly accurate distance measurement and 3D mapping and imaging.

- Direct ToF technology with high sensitivity SPDA detection
- Fast Time-to-Digital Converter (TDC) architecture
- Sub-nanosecond light pulse
- 20 – 2500mm distance sensing @30Hz
- On-chip histogram processing
- 940nm VCSEL Class 1 Eye Safety
- High performance on-chip sunlight rejection filter and algorithm
- Industry's smallest modular OLGA 2.2 x 2.6 x 1.0 mm



Use Cases

There are many different applications where fast accurate ToF measurements can be leveraged . . .

- Auto focus for a mobile phone camera
- Collision avoidance
  - Autonomous vacuum cleaners
  - Industrial robotics
- Presence detection
  - Intrusion detection
  - Auto wake
  - Smart garage
- Smart Warehouse
- Smart Retail, see Avnet Demo Booth
- Autonomous drone landing system
- What use cases can you think of?

Lab Objectives

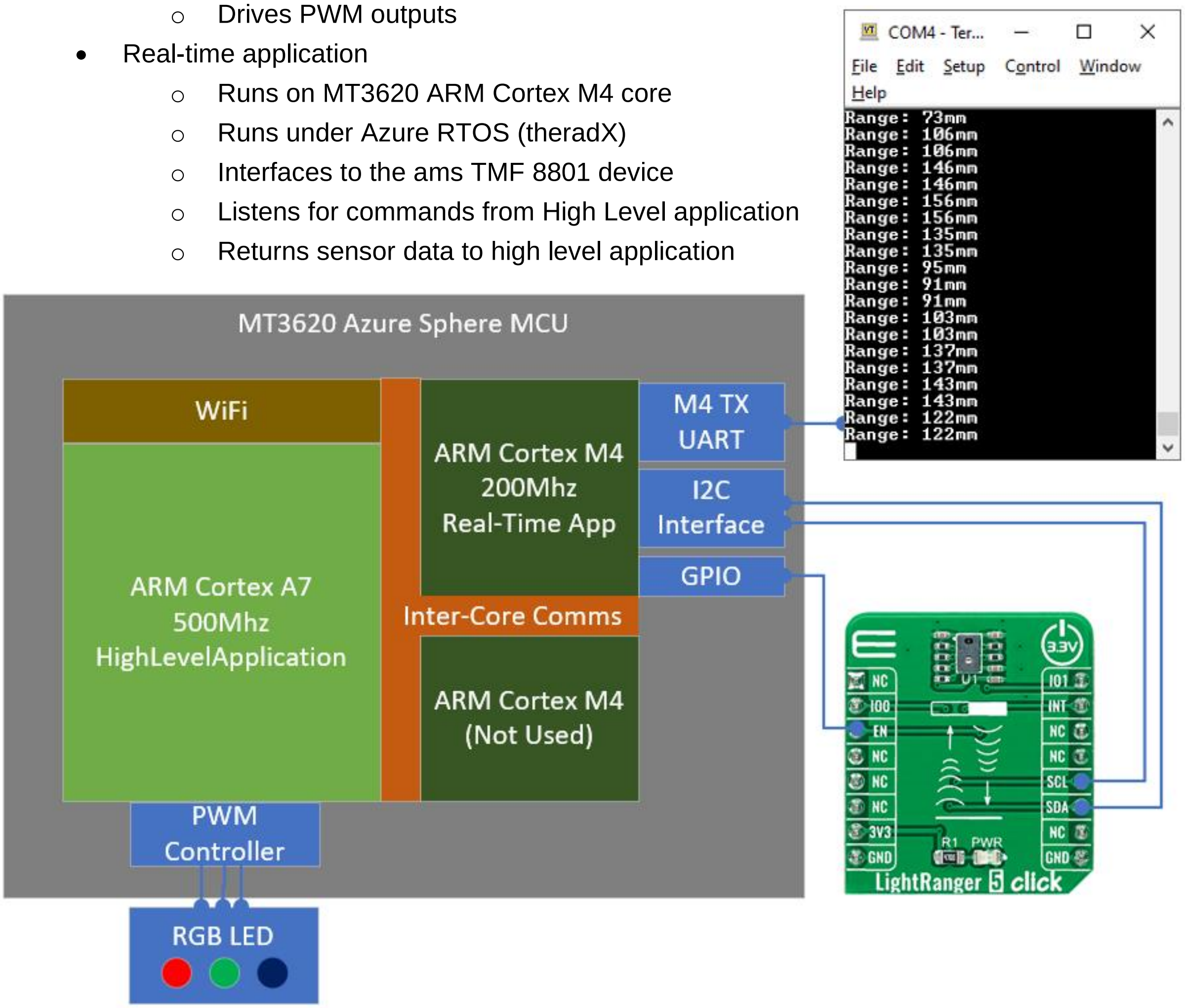
- Learn about the ams TFM 8801
- Configure an Avnet Azure Sphere Starter Kit with a TFM 8801 device
- Install pre-built applications onto the Starter Kit
- Interact with the TFM 8801 sensor

Prerequisites

- Avnet Azure Sphere Starter Kit
- MikroE Lightranger3 Click board
- Azure Sphere SDK installed on laptop
    - Windows or Linux

Demo Architecture

- High Level application
    - Runs on MT3620 ARM Cortex A7 core
    - Runs under the Azure Sphere Linux OS
    - Periodically requests ToF data from the RTApp
    - Drives PWM outputs
- Real-time application
    - Runs on MT3620 ARM Cortex M4 core
    - Runs under Azure RTOS (theradX)
    - Interfaces to the ams TMF 8801 device
    - Listens for commands from High Level application
    - Returns sensor data to high level application



```
Range:  73mm
Range:  106mm
Range:  106mm
Range:  146mm
Range:  146mm
Range:  156mm
Range:  156mm
Range:  135mm
Range:  135mm
Range:  95mm
Range:  91mm
Range:  91mm
Range:  103mm
Range:  103mm
Range:  137mm
Range:  137mm
Range:  143mm
Range:  143mm
Range:  122mm
Range:  122mm
```

Application Details

The example we'll run today consists to two different applications running on an MT3620 Azure Sphere MCU. The applications communicate through an inter-core communication path provided by the Azure Sphere SDK/OS.

Real-Time Application

The application running on the M4 core is an Azure RTOS (threadX) application. The real-time application code can be found on GitHub here, under /Examples/AvnetLightRanger5-RTApp/.

Application pseudocode:

```
•   Initialize hardware
•   Initialize Inter-Core communication path
•   Wait for high level application command
        o   Read sensor
        o   Return Data to high level application
        o   Output range data to M4 UART
```

High-Level Application

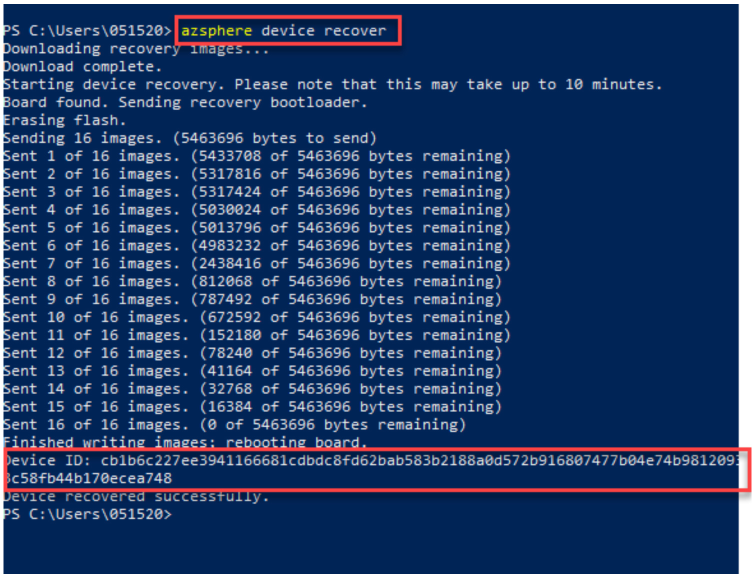
The application running on the A7 High Level core is a POSIX application running on the Azure Sphere Linux OS. The high-level application code can be found on GitHub here.

Application pseudocode:

```
•   Initialize hardware
•   Initialize global variables
        o   Low limit – Defines the range for the Red LED
                ▪   0mm – Low Limit (100mm default)
        o   Medium limit – Defines the range for the Blue LED
                ▪   Low Limit – Medium Limit (200mm default)
        o   High limit – Define the range for the Green LED
                ▪   Medium Limit – High Limit (300mm default)
•   While (true)
        o   Send command to real-time app to read sensor
        o   Catch the sensor data
                ▪   Determine which range the data represents (low, medium, high)
                ▪   Determine where in the current range the new data sits
                ▪   Set the PWM value to drive the brightness of the LED in the range
                        •   Low limit LED bright
                        •   High limit LED dim
```

Lab Instructions

1. Download the workshop package from https://avnet.me/sensorConvergeSphereWorkshop
2. Unpack your new Avnet Azure Sphere Starter Kit
3. Unpack the lightranger3 Click board
    a. Insert the lightrange3 Click board into Mikro BUS location #2 with the TFM8801 sensor closest to the edge of the Starter Kit
4. Connect the Starter Kit to your development PC using the provided micro-USB cable
5. Update your device to the latest OS
    a. Open a command shell
    b. Enter the recover command > **azsphere device recover**
    c. Verify that the images are updated and the device ID is read after the board reboots

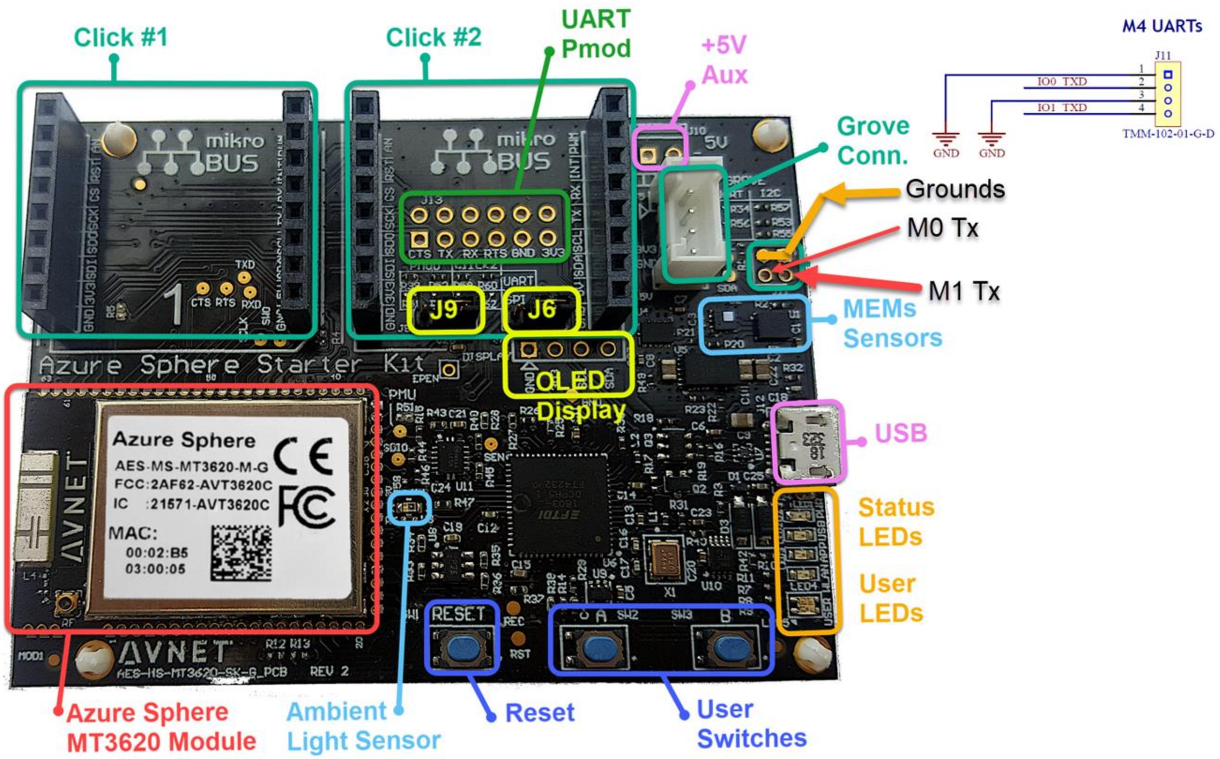


```
PS C:\Users\051520> azsphere device recover
Downloading recovery images...
Download complete.
Starting device recovery. Please note that this may take up to 10 minutes.
Board found. Sending recovery bootloader.
Erasing flash.
Sending 16 images. (5463696 bytes to send)
Sent 1 of 16 images. (5433708 of 5463696 bytes remaining)
Sent 2 of 16 images. (5317816 of 5463696 bytes remaining)
Sent 3 of 16 images. (5317424 of 5463696 bytes remaining)
Sent 4 of 16 images. (5030024 of 5463696 bytes remaining)
Sent 5 of 16 images. (5013796 of 5463696 bytes remaining)
Sent 6 of 16 images. (4983232 of 5463696 bytes remaining)
Sent 7 of 16 images. (2438416 of 5463696 bytes remaining)
Sent 8 of 16 images. (812068 of 5463696 bytes remaining)
Sent 9 of 16 images. (787492 of 5463696 bytes remaining)
Sent 10 of 16 images. (672592 of 5463696 bytes remaining)
Sent 11 of 16 images. (152180 of 5463696 bytes remaining)
Sent 12 of 16 images. (78240 of 5463696 bytes remaining)
Sent 13 of 16 images. (41164 of 5463696 bytes remaining)
Sent 14 of 16 images. (32768 of 5463696 bytes remaining)
Sent 15 of 16 images. (16384 of 5463696 bytes remaining)
Sent 16 of 16 images. (0 of 5463696 bytes remaining)
Finished writing images; rebooting board.
Device ID: cb1b6c227ee3941166681cdbdc8fd62bab583b2188a0d572b916807477b04e74b981209
3c58fb44b170ecea748
Device recovered successfully.
PS C:\Users\051520>
```

    d. If the command times out and fails to read the Device ID
        i. See the troubleshooting documentation here.

6. Connect your USB Serial Port to the board, this will allow us to monitor debug from the AzureRTOS application, use the M1 TX signal closest to the edge for the RX pin on your USB cable.

Signal	Serial Port Cable	Starter Kit
M4 Transmit Data	White (RX)	M1 (TX)
Ground	Black (GRD)	Any valid ground on the Starter Kit



7. Open a serial port terminal application such as TeraTerm or Putty
    a. Set the serial port to 115,200, 8, N, 1
    b. Set the terminal emulation to VT100

8. Sideload the pre-built applications onto the device

    a. Open a command shell such as the Windows PowerShell

    b. Open the folder where you copied the workshop files

    c. Change to the TMF8801 folder

        i. **cd TMF8801**

    d. Copy the real-time application to the device

        i. > **azsphere device sideload deploy --image AvnetLightRanger5Click-Rev2-Click2-Workshop-V1-signed.imagepackage**

    e. Copy the high-level application to the device

        i. > **azsphere device sideload deploy --image avnet_sensors_converge_TMF8801_HLApp-V1-signed.imagepackage**

    f. If you loaded the high level app first, hit the Starter Kit reset button

9. Exercise the sensor

    a. Place your hand or some other object above the Lightranger5 click board

    b. Observe the range measurement displayed as debug in your serial port terminal

    c. Observe the RGB LED on the starter kit identify the range (Red: close, Blue: medium, or Green: far).

    d. Observe that you can vary the LED brightness by moving to either end of any range

Questions?

Additional Resources

- AMS TFM 8801 Datasheet
- Lightranger5 Click board documentation
- Azure Sphere Training Plan
    - Videos and training materials to bring a developer up to speed with the Azure Sphere solution and Azure Sphere Software Development

REVISION HISTORY

Revision	Date	Note
V1	3/28/2022	Initial document